Vehicle scheduling

Pramesh Kumar

IIT Delhi

October 30, 2024

Introduction

Definition (Vehicle scheduling (Schedule Blocking)). Given a set of timetabled transit trips T and a set of transit vehicles V (possibly with varying capacities), find the assignment of trips to vehicle such as each trip in T is assigned to only one vehicle in V and spatial and temporal constraints of serving trips by any given vehicle is satisfied.

- In this step, we break down a schedule in form of blocks, which are the trips assigned to various vehicles to serve during a day.

Definitions (TCRP 135)

Definition (Block). A vehicle (or train) assignment that includes the series of trips operated by each vehicle from the time it pulls out to the time it pulls in.

Definition (Blocking). The process in which trips are hooked together to form a vehicle assignment or block.

Definition (Hooking). The process of attaching the end of a trip in one direction to the beginning of a trip the other direction.

Definition (Interlining). The use of the same vehicle on a block operating on more than one route with the same operator, without returning to the garage during route changes.

Definition (Through-routing). A form of interlining in which a vehicle switches from inbound service on one route to outbound service on another route while continuing in service throughout the day.

Simple blocking exercise

Final output should look like this...

			Eastbound					Westbound					
Block #	**Pull Out**	**A**	**B**	**C**	**D**		**D**	**C**	**B**	**A**		**Next Trip**	**Pull In**
9703	05:55						06:15	06:26	06:40	06:48		07:00	
9701	05:50	06:00	06:08	06:22	06:33		06:45	06:56	07:10	07:18		07:30	
9702	06:20	06:30	06:38	06:52	07:03		07:15	07:26	07:40	07:48		08:00	
9703		07:00	07:08	07:22	07:33		07:45	07:56	08:10	08:18		08:30	
9701		07:30	07:38	07:52	08:03		08:15	08:26	08:40	08:48		09:00	
9702		08:00	08:08	08:22	08:33		08:45	08:56	09:10	09:18		09:30	
9703		08:30	08:38	08:52	09:03		09:15	09:26	09:40	09:48		10:00	
9701		09:00	09:08	09:22	09:33		09:45	09:56	10:10	10:18		10:30	
9702		09:30	09:38	09:52	10:03		10:15	10:26	10:40	10:48		11:00	
9703		10:00	10:08	10:22	10:33		10:45	10:56	11:10	11:18		11:30	
9701		10:30	10:38	10:52	11:03		11:15	11:26	11:40	11:48		12:00	
9702		11:00	11:08	11:22	11:33		11:45	11:56	12:10	12:18		12:30	
9703		11:30	11:38	11:52	12:03		12:15	12:26	12:40	12:48		13:00	
9701		12:00	12:08	12:22	12:33		12:45	12:56	13:10	13:18		13:30	
9702		12:30	12:38	12:52	13:03		13:15	13:26	13:40	13:48		14:00	
9703		13:00	13:08	13:22	13:33		13:45	13:56	14:10	14:18		14:30	
9701		13:30	13:38	13:52	14:03		14:15	14:26	14:40	14:48		15:00	
9702		14:00	14:08	14:22	14:33		14:45	14:56	15:10	15:18		15:30	
9703		14:30	14:38	14:52	15:03		15:15	15:26	15:40	15:48		16:00	
9701		15:00	15:08	15:22	15:33		15:45	15:56	16:10	16:18		16:30	
9702		15:30	15:38	15:52	16:03		16:15	16:26	16:40	16:48		17:00	
9703		16:00	16:08	16:22	16:33		16:45	16:56	17:10	17:18		17:30	
9701		16:30	16:38	16:52	17:03		17:15	17:26	17:40	17:48		18:00	
9702		17:00	17:08	17:22	17:33		17:45	17:56	18:10	18:18		18:30	
9703		17:30	17:38	17:52	18:03		18:15	18:26	18:40	18:48		19:00	
9701		18:00	18:08	18:22	18:33		18:45	18:56	19:10	19:18			19:28
9702		18:30	18:38	18:52	19:03		19:15	19:26	19:40	19:48			19:58
9703		19:00	19:08	19:22	19:33								19:53

Figure: Each color represents a block[1]

[1]Source: TCRP135

Optimization approaches

Depending on the number of depots, we can classify the vehicle scheduling into two types:

- Single-depot vehicle scheduling problem (SD-VSP)
- Multi-depot vehicle scheduling problem (MD-VSP)

Single-depot vehicle scheduling

Minimal decomposition by Jahar Saha (1970)

Let T be the set of trips and a relation between any two trips $i \ \alpha \ j$ which tells that whether trip j can be served after i by the same vehicle, i.e., $i \ \alpha \ j$ if

- j starts at the same station as i
- j's dispatching time is later then i's arrival time.

Define $C_{ij} = \begin{cases} 1 & \textbf{if } i \ \alpha \ j \\ -\infty & \textbf{otherwise} \end{cases}$

Minimal decomposition by Jahar Saha (1970)

Decisions: $X_{ij} = 1$ if j is performed right after i

$$\underset{\mathbf{x}}{\text{maximize}} \quad \sum_{i \in T} \sum_{j \in T} C_{ij} X_{ij} \tag{1a}$$

$$\text{subject to} \quad \sum_{j \in T} x_{ij} \leq 1, \forall i \in T \tag{1b}$$

$$\sum_{i \in T} x_{ij} \leq 1, \forall j \in T \tag{1c}$$

$$X_{ij} = \{0, 1\}, \forall i \in T, \forall j \in T \tag{1d}$$

Remark.

- It can be solved as max flow problem using Ford and Fulkerson algorithm.
- Final values of x_{ij} are used to find the vehicle blocks.
- It gives minimum fleet for serving the given number of trips.
- The formulation has following disadvantages:
  - Cost of operation not considered

Assignment formulation by Freling et al. (2001)

Let b_i and e_i be the starting and ending locations, and let bt_i and et_i be the starting and ending times of a trip i respectively.

Definition (Compatible trips). Two trips i and j are said to be compatible pair of trips if the same vehicle can cover these trips in sequence, i.e., if $et_i + t(e_i, b_j) \leq bt_j$, where $t(e_j, b_i)$ is the deadheading travel time from e_j to location b_i.

Let T be the set of trips and $E = \{(i,j) : i \text{ and } j \text{ are compatible}\}$ be the set of compatible trips.

Consider a graph $G(M, E^T)$, where $M = T \cup \{|T|+1, |T|+2, \cdots, 2|T|\}$ and $E^T = E \cup \{(i, |T|+i) \mid i = 1, \cdots |T|\}$.

Remark. A path of the form $\{i_1, \cdots, i_k, |T| + i_k\}$ is to a feasible vehicle schedule leaving the depot to perform trips $i_1, \cdots, i_k$ and returning to the depot afterwards.

Let c_{ij} be the cost of serving j right after i (deadhead time). Further c_{sj} and c_{it} be the deadhead cost of going from depot to start of trip j and deadhead cost of going from end location of trip i to depot respectively.

Assignment formulation by Freling et al. (2001)

Remark. If a vehicle is serving j after i, then it will not have to go to j from depot. Based on that, let us define

$$b_{ij} = \begin{cases} c_{ij} - c_{sj}, & \forall (i,j) \in E \\ c_{it}, & \forall (i,j) \in E^T \backslash E \end{cases}$$

Decisions: $y_{ij} = 1$ if j is performed right after i

$$\begin{aligned}
\underset{\mathbf{y}}{\text{minimize}} \quad & \sum_{(i,j)\in E^T} b_{ij} y_{ij} + \sum_{j \in N} c_{sj} && \text{(2a)} \\
\text{subject to} \quad & \sum_{j:(i,j)\in E^T} y_{ij} = 1, \forall i \in T && \text{(2b)} \\
& \sum_{i:(i,j)\in E^T} y_{ij} \leq 1, \forall j \in M && \text{(2c)} \\
& y_{ij} = \{0,1\}, \forall (i,j) \in E^T && \text{(2d)}
\end{aligned}$$

Formulation by L. Bodin (1983)

Consider a digraph $G = (N, A)$ where $N = T \cup \{s, t\}$, where s and t represent the depot and $A = \{(s, i) \mid i \in T\} \cup \{(j, t) \mid j \in T\} \cup \{(i, j) : i, j \in T \text{ and } e_i^t + d_{ij} \leq b_j^t\} \cup \{(t, s)\}$. Here, d_{ij} is the deadhead time between e_i and b_j.

Define $c_{ij} = \begin{cases} d_{ij}, & \forall (i,j) \in A : i, j \in T \\ d_{si}, & \forall (s,i) : i \in T \\ d_{jt}, & \forall (j,t) : j \in T \\ c_0, & i = t \text{ and } j = s \end{cases}$,

where c_0 is the cost of adding extra vehicle. Also, assume p is the maximum number of vehicles allowed.

Decisions: x_{ij} = flow on arc (i, j)

$$
\begin{aligned}
\underset{\mathbf{x}}{\text{minimize}} \quad & \sum_{(i,j)\in A} c_{ij}x_{ij} && \text{(3a)} \\
\text{subject to} \quad & \sum_{i:(i,j)\in A} x_{ij} - \sum_{i:(j,i)\in A} x_{ji} = 0, \forall j \in N && \text{(3b)} \\
& \sum_{i:(i,j)\in A} x_{ij} = 1, \forall j \in T && \text{(3c)} \\
& x_{ts} \leq p && \text{(3d)} \\
& x_{ij} \in \mathbb{Z}_+, \forall (i, j) \in A && \text{(3e)}
\end{aligned}
$$

Multi-depot vehicle scheduling

Formulation by L. Bodin (1983)

Let K be the set of depots. Each depot k has stationed p_k vehicles. We construct $|K|$ directed graphs $G_k = (N_k, A_k)$, where $N_k = T \cup \{s_k\} \cup \{t_k\}$ and $A_k = \{(s_k, i) \mid i \in T\} \cup \{(j, t_k) \mid j \in T\} \cup \{(i,j) : i, j \in T \text{ and } e_i^t + d_{ij} \leq b_j^t\} \cup \{(t_k, s_k)\}$. Costs are defined as before.

Decisions: $x_{ij}^k =$ flow on arc $(i,j) \in A_k$

$$\underset{\mathbf{x}}{\text{minimize}} \quad \sum_{k \in K} \sum_{(i,j) \in A_k} c_{ij}^k x_{ij}^k \tag{4a}$$

$$\text{subject to} \quad \sum_{i:(i,j) \in A_k} x_{ij}^k - \sum_{i:(j,i) \in A_k} x_{ji}^k = 0, \forall j \in N_k, \forall k \in K \tag{4b}$$

$$\sum_{k \in K} \sum_{i:(i,j) \in A} x_{ij}^k = 1, \forall j \in T \tag{4c}$$

$$\sum_{j:(s_k,j) \in A_k} x_{s_k j} \leq p_k, \forall k \in K \tag{4d}$$

$$x_{ij}^k \in \mathbb{Z}_+, \forall (i,j) \in A \tag{4e}$$

This is NP-Hard. Students are encouraged to read about MDVSP with time windows.

Suggested reading

- Gkiotsalitis, Konstantinos. Public transport optimization, Chapter 11.
- TCRP Report 135
- Saha, J. L. "An algorithm for bus scheduling problems." Journal of the Operational Research Society 21.4 (1970): 463-474.
- Freling, Richard, Albert PM Wagelmans, and José M. Pinto Paixão. "Models and algorithms for single-depot vehicle scheduling." Transportation Science 35.2 (2001): 165-180.
- Bodin, Lawrence. "Routing and scheduling of vehicles and crews." Computer & Operations Research 10.2 (1983): 69-211.

Thank you!